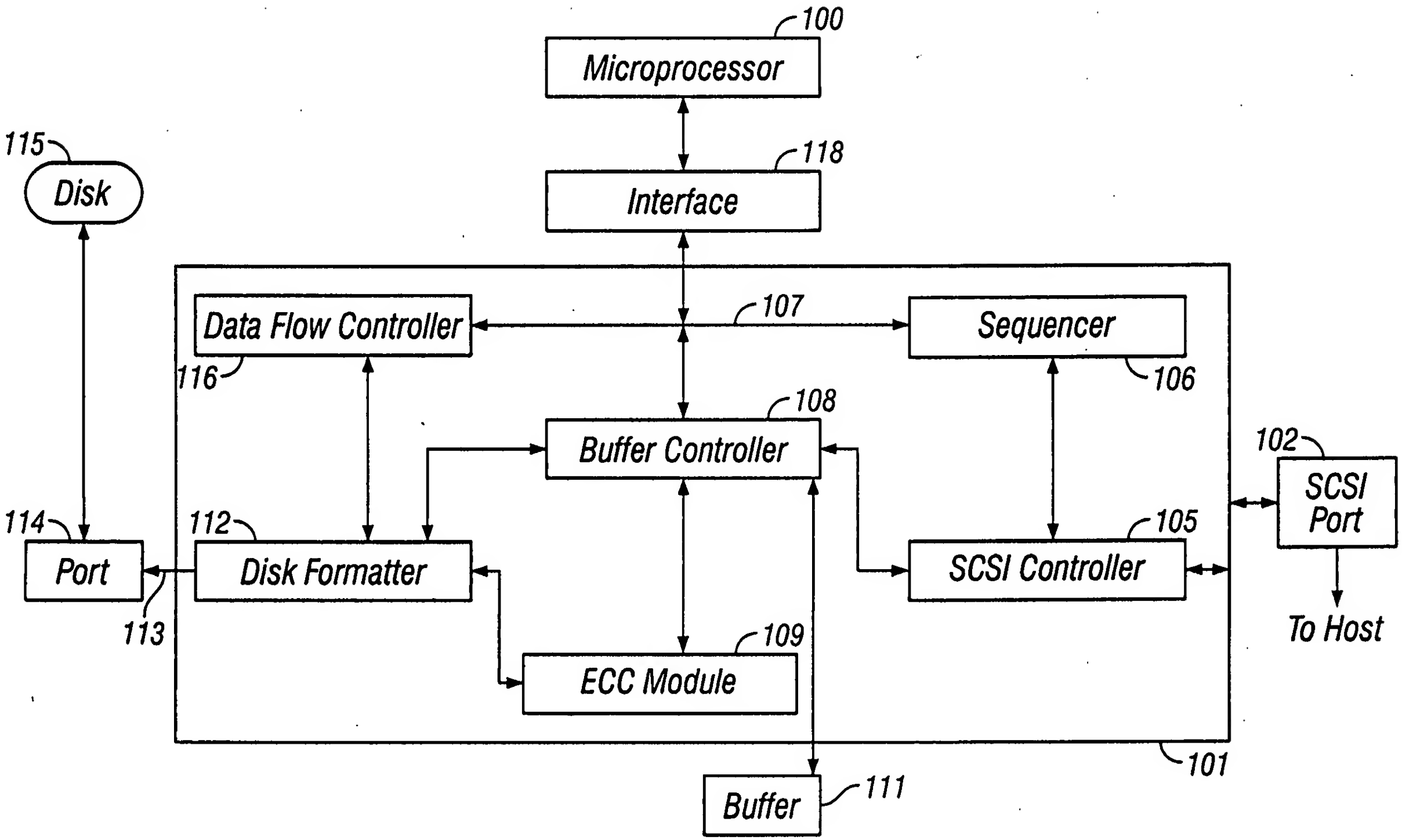


FIG. 1
(Prior Art)

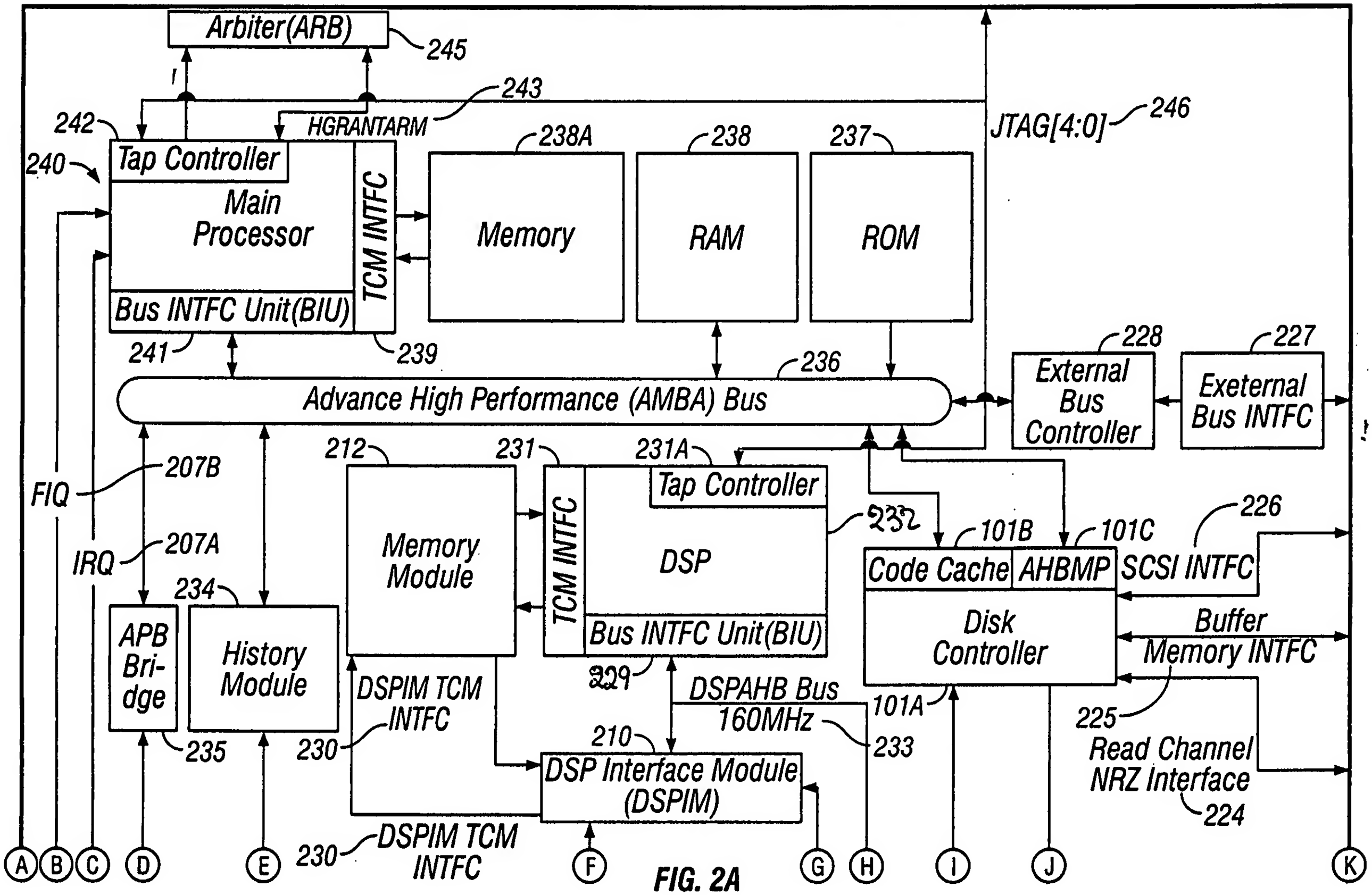

Arbiter(ARB)
245
243
HGRANTARM
242
240
Tap Controller
Main Processor
Bus INTFC Unit(BIU)
TCM INTFC
238A
Memory
238
RAM
237
ROM
JTAG[4:0]
246
241
239
236
Advance High Performance (AMBA) Bus
228
External Bus Controller
227
Exeternal Bus INTFC
212
231
231A
207B
FIQ
207A
IRQ
234
Memory Module
TCM INTFC
Tap Controller
DSP
Bus INTFC Unit(BIU)
232
101B
101C
226
Code Cache
AHBMP
SCSI INTFC
APB Bri-dge
History Module
229
DSPIM TCM INTFC
DSPAHB Bus 160MHz
Disk Controller
Buffer Memory INTFC
101A
225
235
230
210
233
DSP Interface Module (DSPIM)
Read Channel NRZ Interface
224
230
DSPIM TCM INTFC
A
B
C
D
E
F
G
H
I
J
K

FIG. 2A

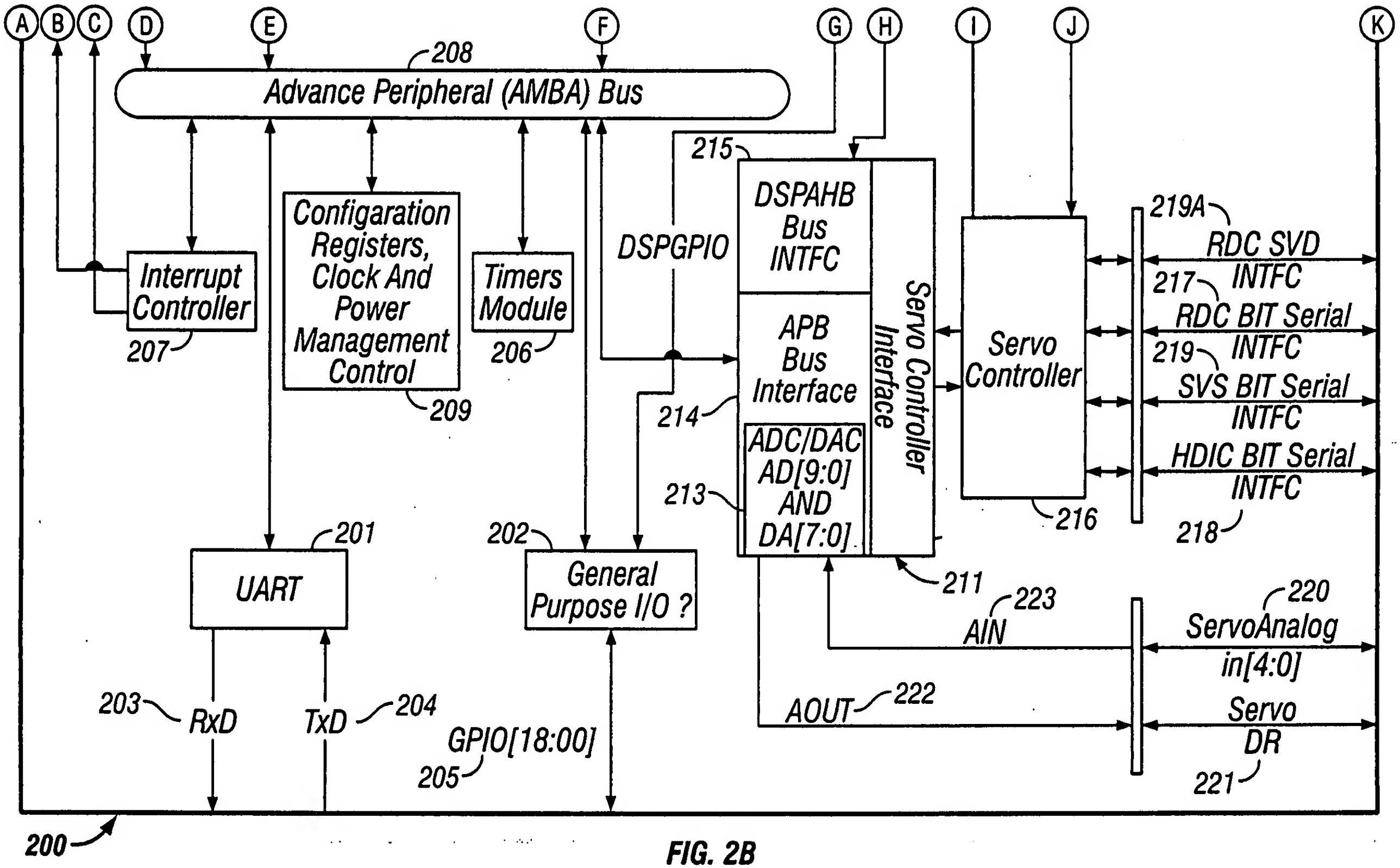

A
B
C
D
E
F
G
H
I
J
K
208
Advance Peripheral (AMBA) Bus
Interrupt Controller
207
Configaration Registers, Clock And Power Management Control
209
Timers Module
206
DSPGPIO
215
DSPAHB Bus INTFC
APB Bus Interface
214
ADC/DAC AD[9:0] AND DA[7:0]
213
Servo Controller Interface
Servo Controller
216
219A
RDC SVD INTFC
217
RDC BIT Serial INTFC
219
SVS BIT Serial INTFC
HDIC BIT Serial INTFC
218
211
223
AIN
220
ServoAnalog in[4:0]
AOUT
222
Servo DR
221
UART
201
202
General Purpose I/O ?
203
RxD
TxD
204
GPIO[18:00]
205
200

FIG. 2B

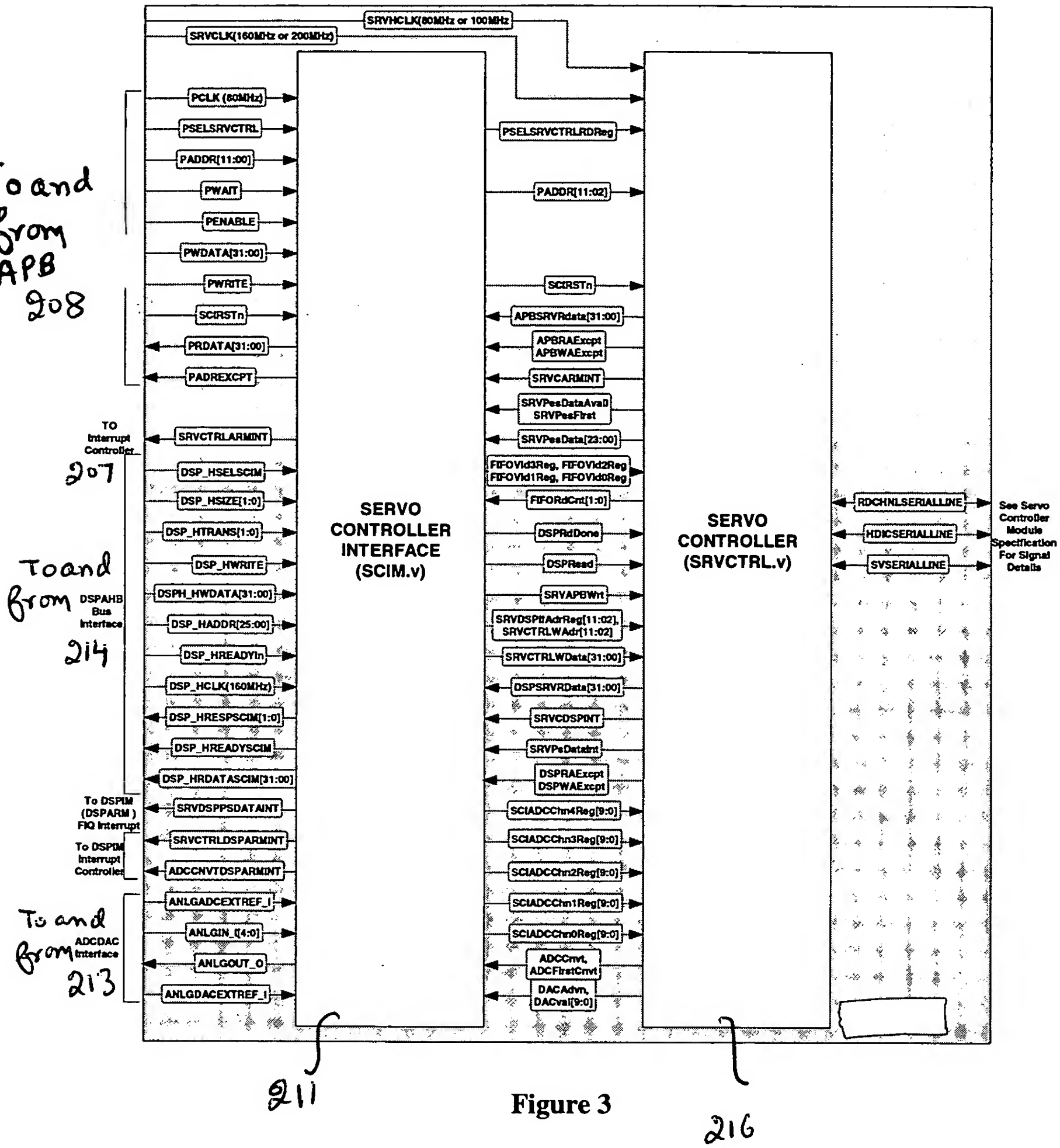

SRVHCLK(80MHz or 100MHz
SRVCLK(160MHz or 200MHz)
PCLK (80MHz)
PSELSRVCTRL
PADDR[11:00]
PWAIT
PENABLE
PWDATA[31:00]
PWRITE
SCIRSTn
PRDATA[31:00]
PADREXCPT
TO Interrupt Controller
SRVCTRLARMINT
DSP_HSELSCIM
DSP_HSIZE[1:0]
DSP_HTRANS[1:0]
DSP_HWRITE
DSPH_HWDATA[31:00]
DSP_HADDR[25:00]
DSP_HREADYIn
DSP_HCLK(160MHz)
DSP_HRESPSCIM[1:0]
DSP_HREADYSCIM
DSP_HRDATASCIM[31:00]
To DSPIM (DSPARM) FIQ Interrupt
SRVDSPPSDATAINT
To DSPIM Interrupt Controller
SRVCTRLDSPARMINT
ADCCNVTDSPARMINT
ANLGADCEXTREF_I
ANLGIN_I[4:0]
ANLGOUT_O
ANLGDACEXTREF_I
SERVO CONTROLLER INTERFACE (SCIM.v)
PSELSRVCTRLRDReg
PADDR[11:02]
SCIRSTn
APBSRVRdata[31:00]
APBRAExcpt
APBWAExcpt
SRVCARMINT
SRVPesDataAvail
SRVPesFirst
SRVPesData[23:00]
FIFOVld3Reg, FIFOVld2Reg
FIFOVld1Reg, FIFOVld0Reg
FIFORdCnt[1:0]
DSPRdDone
DSPRead
SRVAPBWrt
SRVDSPItAdrReg[11:02],
SRVCTRLWAdr[11:02]
SRVCTRLWData[31:00]
DSPSRVRData[31:00]
SRVCDSPINT
SRVPesDataInt
DSPRAExcpt
DSPWAExcpt
SCIADCChn4Reg[9:0]
SCIADCChn3Reg[9:0]
SCIADCChn2Reg[9:0]
SCIADCChn1Reg[9:0]
SCIADCChn0Reg[9:0]
ADCCnvt,
ADCFirstCnvt
DACAdvn,
DACval[9:0]
SERVO CONTROLLER (SRVCTRL.v)
RDCHNLSERIALLINE
HDICSERIALLINE
SVSERIALLINE
See Servo Controller Module Specification For Signal Details
To and from DSPAHB Bus Interface 214
To and from ADCDAC Interface 213
211
216
207

Figure 3

Signal	Count	Description
PCLK	1	**APB 208 Clock signal** controls register loads and register reads associated with APB 208 transactions. All action in SCI 211 associated with APB 208 occurs on the low to high assertion of the PCLK signal.
SCIRSTn	1	**APB 208 Reset or firmware reset** signal is asserted **active low** when APB 208 is reset or when the firmware sets the MdlRstReg[07]. When this signal is asserted all registers in SCI 211 and SC 216 are cleared to a default reset value and interrupts are disabled.
PENABLE	1	**APB 208 Enable** signal is asserted high active for 1 clock cycle by APB Bridge 235 to enable all register access associated with APB interface 214 in SCI 211.
PSELSRVCTRL	1	**This** signal when asserted high indicates that APB Bridge235 has selected SCI 211 for an APB 208 transaction. One signal is provided for each APB 208 peripheral. This signal is asserted by APB Bridge 235 at the start of the transaction and remains asserted until the transaction is completed.
PADDR[11:00]	12	**This is an APB 208 Address** (for example, 12 bits) that is decoded by SCI 211 to select a local register or register during an APB Bridge 235 access. If the address is for SC 216, then SCI 211 passes this address to SC 216. SCI 211 also validates the address. These signals are asserted by APB Bridge 235 at the start of the transaction and remains asserted until the transaction completes.
PWRITE	1	**This is an APB208 Write** signal that indicates a write transaction when asserted high and a read transaction when asserted low. This signal is asserted by APB Bridge 235 at the beginning of the transaction and remains asserted until the transaction is completed.
PWDATA[31:00]	32	**These are APB 208 Write Data** signals that are driven by APB Bridge 235 and when executing a write transaction provides the associated write data for these signals.

Figure 4A

Signal	Count	Description
PRDATA[31:00]	32	**The APB 208 Read Data Signal**:: When APB Bridge 235 selects SCI 211 and the PWRITE signal **is asserted low,** SCI 211 supplies the read data based on the decode of the PADDR[11:00]. When the read data is from SC 216, SC 211 synchronizes this signal to the 80MHz clock domain and drives bits 31:00 of APB 208 read data.
SRVCTRLARMINT	1	**This is an Interrupt from SCI 211 to the Interrupt Controller 207**: SCI 211 synchronizes this signal to the 80MHz clock domain.
PWAIT	1	**This APB 208 Bus Wait signal** is asserted by SCI 211 to indicate to APB Bridge235 that it cannot complete the requested read transaction in two bus cycles. To insert a wait state this signal is asserted before APB Bridge 235 asserts PENABLE. When the signal is de-asserted (synchronous with PCLK), APB Bridge 235 asserts PENABLE on the subsequent PCLK.
PADREXCPT	1	**This indicates an APB 208 Address Exception.** It is asserted high for one PCLK clock cycle when SCI 211 detects an APB 208 "address exception".

Figure 4B

Signal	Count	Description
DSP_HCLK	1	**This is the DSPAHB Bus 233 clock** signal that controls register loads; status reads and synchronizes bus transactions to DSP 232 (that acts a bus-master). All action with the DSPAHB Bus 233 in SCI 211 occurs on low to high assertion of this clock.
DSP_HSELSCIM	1	**This** signal is asserted by DSPAHB Bus 233 Address Decoder Module decoding the most significant 6 bits of the DSPAHB Address (DSP_HADDR[31:26]) which selects the assigned address space, for SCI 211 and SC 216.
DSP_HTRANS[1:0]	2	**This DSPAHB Bus Transfer signal** indicates the type of transfer a current DSPAHB Bus 233 transaction is executing (NONSEQUENTIAL, SEQUENTIAL, IDLE or BUSY). SCI 216 uses this information to determine the transfer type and the Response (AHBHRESP[1:0]) that will be returned to Bus 233.
DSP_HWRITE	1	**This DSPAHB Write** signal specifies whether the DSPAHB Bus 233 transaction is a read (DSP_HWRITE = 0) access or a write (DSP_HWRITE = 1) access.
DSP_HADDR[25:00]	26	**This signal indicates the DSPAHB Address.** The 26 bits of the DSPAHB Bus 233 address specify the memory mapped register 1209 or a memory location in SCI 211 or SC 216 bus master (DSP 232) is accessing.
DSP_HWDATA[31:00]	32	**The DSPAHB Write Data** [31:00] signal transfers the data from the DSPAHB Bus 233 master to SCI 211 during a write (store) transaction. SCI 211 registers these signals and either writes to a local register with this data or transfers the data to SC 216.
DSP_HSIZE[1:0]	2	**The DSPAHB Bus Size** signal specifies the size of a transfer, i.e., byte, half word, or word. Byte = 00b: Halfword = 01b: Word = 10b: Invalid = 11b.
DSP_HREADYin	1	**The DSPAHB Bus Ready In signal** indicates that a DSPAHB Bus 233 transaction has completed and a new transaction is starting. SCI 211 uses the assertion of this signal to enable loading the address and control information (DSPAHB Address, DSPAHB Size, DSPAHB Transfer, DSPAHB Write and DSPHSELSRVCTRL).

Figure 5A

Signal	Count	Description
DSP_HRDATASCIM[31:00]	32	**The Servo Controller Interface Read Data** signals are used to transfer read data from SCI 211 to the requesting bus master (DSP 232) through the Slave To Master Multiplexer (not shown) of the DSPAHB Bus 233.
DSP_HRESPSCIM[1:0]	2	**This signal provides a Servo Controller Interface Response.** When SCI 211 is selected to respond to a bus master (DSP 232) then SCI 211 indicating the status of the requested bus transaction it has completed controls the signals. These signals are returned to the bus master through the Slave To Master Multiplexer (not shown) of the DSPAHB Bus 233.
DSP_HREADYout	1	**Servo Controller Interface HREADYout** signal: When SCI **211** is selected to respond to a bus master(DSP 232) , SCI 211 assets this signal indicating to the bus master that it has completed the requested bus transaction . This signal is returned to the bus master through the Slave To Master Multiplexer of the DSPAHB Bus 233.

Figure 5B

Signal	Count	Description
SRVCTRLDSPARMINT	1	**Servo Controller DSP 232 Interrupt** - This signal, if enabled, is asserted as a level sensitive interrupt to APBDSP Interface (DSPIRQ[0]) 214 indicating that an exception has occurred in SCI 211 or SC 216 due to an action by DSP 232. To determine the source of the exception, the SCIDSPARMStatusReg is interrogated.
SRVDSPPSDATAINT	1	**Servo Controller DSP Positioning Data Available Interrupt** signal if enabled, is asserted as an edge sensitive interrupt (one DSP_HCLK clock cycle pulse) to DSP 232 FIQ input indicating that SC 216 has error positioning information available for processing.
ADCCNVTDSPARMINT	1	**ADC Conversion Complete Interrupt to DSP 232** signal, if enabled is asserted as an edge sensitive interrupt (one DSP_HCLK clock cycle pulse) to the DSP Interrupt Controller in the DSP Interface Module 210 indicating that SCI 211 has ADC (analog to digital) conversion information available for processing.

Figure 6

Signal	Count	Description
PRESETn (directly from APB Bridge 235)	1	**APB 208 Reset** signal is asserted **active low** when APB 208 is reset. When this signal is asserted all registers in SCI 211 and SC 216 are cleared to a default reset value and interrupts are disabled. SC 216 synchronizes this signal to its clock domain.
PSELSRVCTRLRDReg	1	**Servo Controller Read from APB 208** – This signal is asserted by SCI 211 when an APB 208 read transaction is requested. This signal is synchronized by SC 216 to its clock domain and SC 216 based on PADDR[11:02], takes a "snapshot" of the contents of the accessed register or supplies the data of the accessed static register to SCI 211. If SC 216 takes a "snapshot" of the contents of a dynamic register then SC 216 supplies stable data to SCI 211.
PADDR[11:02]	10	**APB Address.** These 10 bits are APB 208 address. SCI 211 decodes these 10 bits to select a local register in SCI 211 or SC 216 register during an APB Bridge235 transaction. If the access is a read transaction SCI 211 passes these signals on to SC 216 to select the contents of the specified register. SC 216 using PSELSRVCTRLRD synchronizes these signals to SC 216 clock domain. These signals remain asserted until the read transaction has completed on APB 208.

Figure 7

Signal	Count	Description
FIFOVld0Reg FIFOVld1Reg FIFOVld2Reg FIFOVld3Reg	4	**Write FIFO Entry x [3:0] Valid Signals** - When SCI 211 asserts one of these signals entry x is valid ready for SC 216 to write the addressed register. SCI 216 asserts this signal for 2 DSP_HCLK (160MHz) clock cycles.
SRVCTRLWAdr[9:0]	10	**FIFO Entry Write Address** specifies the memory mapped register that SCI 211 is writing in SC 216. SCI 211 validates this address before passing it on.
SRVCTRLWData[31:00]	32	**FIFO Entry Write Data** signals are associated with the write data address, SRVCTRLWAdr[9:0], that specifies the memory mapped register 1209 in SC 216 that SCI 211 is writing to.
SRVAPBWrt	1	**Servo APB Write signal** specifies the interface which generated the write entry in the Speed Matching Write FIFO 1206. SC 216 uses this information to route "write address exceptions" back to the initiating interface in SCI 211. SRVAPBWrt = 0 = Auxiliary Register Interface. SRVAPBWrt = 1 = APB write.

Figure 8

Signal	Count	Description
FIFORdCnt[1:0]	2	**FIFO Read Count** signal selects the FIFO (1206) entry that SC 216 is reading the address and writing data from.

Figure 9

Signal	Count	Description
DSPRead	1	**DSP Read access request** - When this signal is asserted (high) DSP 232 is requesting a read of a SC 216 memory mapped register (1209) specified by DSP_HADDR[11:02]. This signal remains asserted until the read transaction is completed.
DSPRdDone	1	**DSP Read Done** – SCI 211 asserts this signal for 1 DSP_HCLK clock cycle indicating that DSP 232 read will complete this cycle.
SRVDSPIfAdrReg[11:02]	10	**DSPARM Interface Address Register** - The contents of this register specifies the address of the memory-mapped register (1206) that DSP 232 is accessing in SC 216. SCI 211 validates this address before passing it on.

Figure 10

Signal	Count	Description
APBSRVRdata[31:00]	32	**APB Servo Controller Read Data** is the read data from SC 216 to SCI 211 based on the decode of the PADDR [11:02] signal. SCI 211 synchronizes this data to the PCLK domain and loads APBSRVRdReg[31:00] which drives PRDATA[31:00] to APB 208.
DSPSRVRdata[31:00]	32	**DSPARM Servo Controller Read Data** is the read data from SC 216 to SCI 211 based on the decode of SRVDSPIfAdrReg[11:00] signal. SCI 211 synchronizes this read data to the DSP_HCLK domain and loads SRVRDataReg[31:00] which drives DSP_HRDATA[31:00] to DSP 232
SRVCARMINT	1	**Servo Controller ARM Interrupt** signal is a level sensitive interrupt from SC 216 when routing an interrupt to processor 240. This signal is synchronized to the SCI clock domain (DSP_HCLK).
SRVCDSPINT	1	**SC 216 when routing an interrupt to DSP 232 asserts this Interrupt signal.** SCI 211 synchronizes this level sensitive signal to the DSP_HCLK domain. This interrupt is asserted for exceptions that are routed to DSP 232 with the exception of the positioning data interrupt (SRVPsDataInt) from the Read Channel.
SRVPsDataInt	1	**SC 216 asserts this Servo Positioning Data Interrupt signal for two clock cycles** when the Servo Mechanism position data is available from the Read Channel. SCI 211 synchronizes this signal to the DSP 232 clock domain (DSP_HCLK) while generating the interrupt.
APBRAExcpt	1	**This APB 208 Read Address Exception signal occurs when SC 216** detects an address exception on an APB 208 read transaction. SC 216 asserts this signal for two (SRVCLK) clock cycles. SCI 211 captures this signal and asserts the PADREXCPT signal to APB Bridge 235.
APBWAExcpt	1	**The APB 208 Write Address Exception occurs when SC 216** detects an address exception on an APB 208 write transaction. SC 216 asserts this signal for two (SRVCLK) clock cycles. SCI 211 captures this signal and asserts the PADREXCPT signal to APB Bridge 235.
DSPRAExcpt	1	**The DSP 232 Read Address Exception occurs when** SC 216 detects an address exception for a DSP 232 read transaction. SC 216 asserts this signal for two (SRVCLK) clock cycles. SCI 211 captures this signal and generates an interrupt to DSP 232.
DSPWAExcpt	1	**DSPARM Write Address Exception** - SC 216 detects an address exception on a DSP 232 write transaction. SC 216 asserts this signal for two (SRVCLK) clock cycles. SCI 211 captures this signal and generates an interrupt to DSP 232.

Figure 11

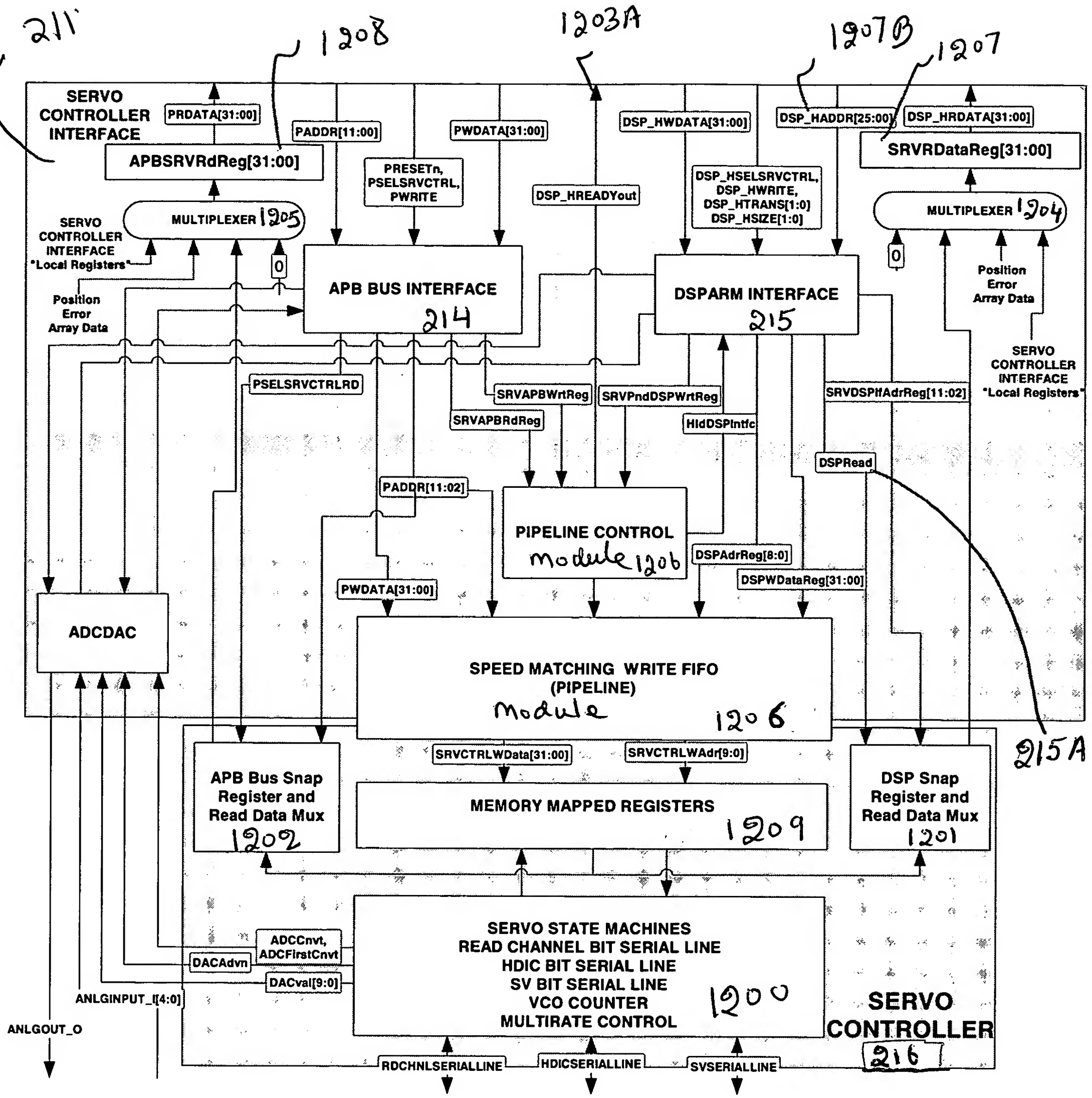

211
1208
1203A
1207B
1207
SERVO CONTROLLER INTERFACE
PRDATA[31:00]
PADDR[11:00]
PWDATA[31:00]
DSP_HWDATA[31:00]
DSP_HADDR[25:00]
DSP_HRDATA[31:00]
APBSRVRdReg[31:00]
SRVRDataReg[31:00]
PRESETn, PSELSRVCTRL, PWRITE
DSP_HREADYout
DSP_HSELSRVCTRL, DSP_HWRITE, DSP_HTRANS[1:0] DSP_HSIZE[1:0]
MULTIPLEXER 1205
MULTIPLEXER 1204
SERVO CONTROLLER INTERFACE "Local Registers"
0
0
Position Error Array Data
Position Error Array Data
APB BUS INTERFACE 214
DSPARM INTERFACE 215
SERVO CONTROLLER INTERFACE "Local Registers"
PSELSRVCTRLRD
SRVAPBWrtReg
SRVPndDSPWrtReg
SRVDSPIfAdrReg[11:02]
SRVAPBRdReg
HldDSPIntfc
DSPRead
PADDR[11:02]
PIPELINE CONTROL module 1206
DSPAdrReg[8:0]
PWDATA[31:00]
DSPWDataReg[31:00]
ADCDAC
SPEED MATCHING WRITE FIFO (PIPELINE) Module 1206
215A
SRVCTRLWData[31:00]
SRVCTRLWAdr[9:0]
APB Bus Snap Register and Read Data Mux 1202
MEMORY MAPPED REGISTERS 1209
DSP Snap Register and Read Data Mux 1201
ADCCnvt, ADCFirstCnvt
DACAdvn
DACval[9:0]
SERVO STATE MACHINES
READ CHANNEL BIT SERIAL LINE
HDIC BIT SERIAL LINE
SV BIT SERIAL LINE
VCO COUNTER
MULTIRATE CONTROL
1200
SERVO CONTROLLER 216
ANLGINPUT_I[4:0]
ANLGOUT_O
RDCHNLSERIALLINE
HDICSERIALLINE
SVSERIALLINE

Figure 12A

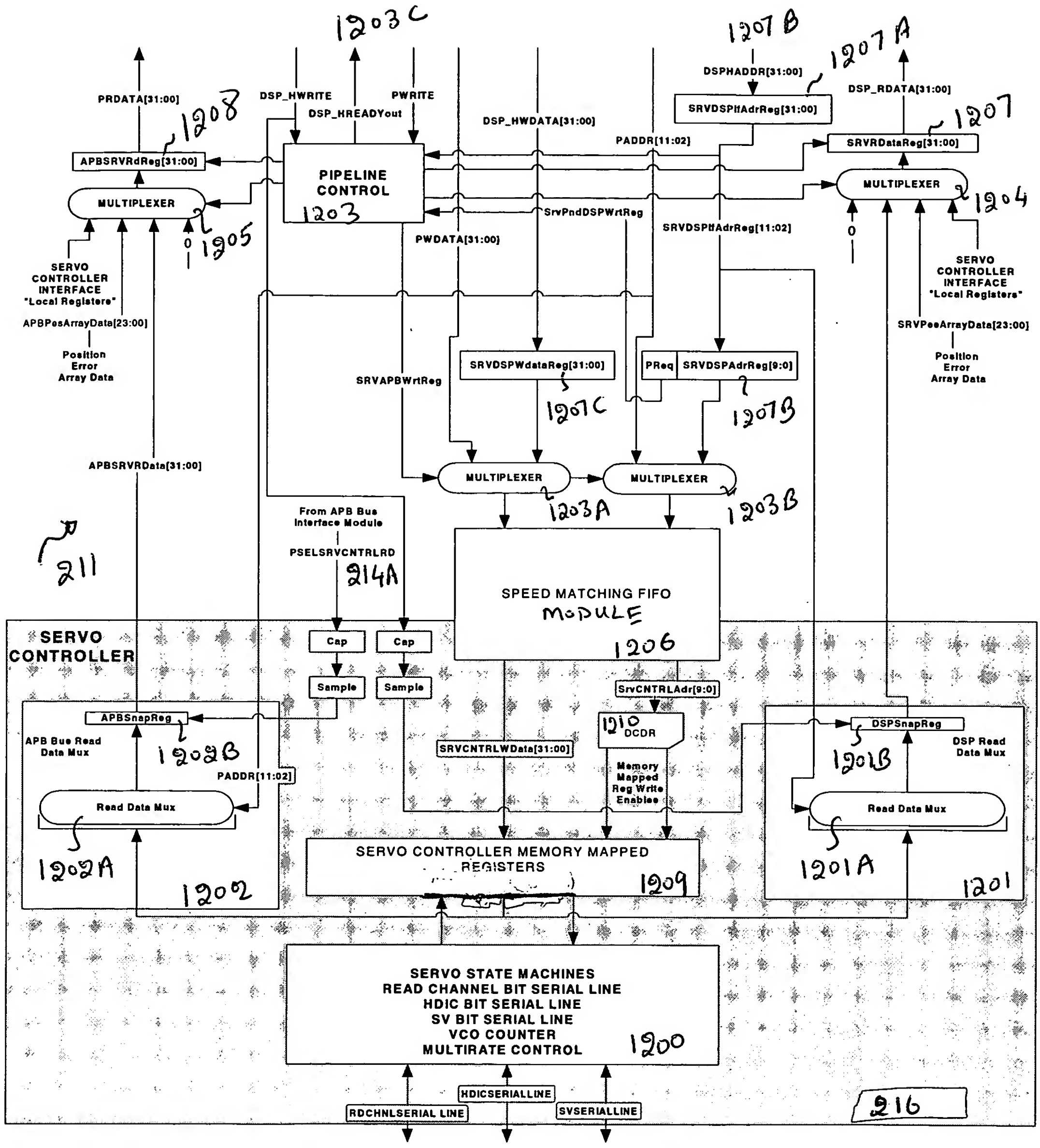

1203C
1207B
1207A
PRDATA[31:00]
1208
DSP_HWRITE
PWRITE
DSP_HREADYout
DSPHADDR[31:00]
SRVDSPHAdrReg[31:00]
DSP_RDATA[31:00]
1207
DSP_HWDATA[31:00]
PADDR[11:02]
SRVRDataReg[31:00]
APBSRVRdReg[31:00]
PIPELINE CONTROL
1203
MULTIPLEXER
1204
MULTIPLEXER
1205
SrvPndDSPWrtReg
SRVDSPHAdrReg[11:02]
PWDATA[31:00]
0
0
SERVO CONTROLLER INTERFACE "Local Registers"
SERVO CONTROLLER INTERFACE "Local Registers"
APBPosArrayData[23:00]
SRVPosArrayData[23:00]
Position Error Array Data
Position Error Array Data
SRVDSPWdataReg[31:00]
PReq
SRVDSPAdrReg[9:0]
SRVAPBWrtReg
1207C
1207B
APBSRVRData[31:00]
MULTIPLEXER
MULTIPLEXER
1203A
1203B
From APB Bus Interface Module
PSELSRVCNTRLRD
214A
211
SPEED MATCHING FIFO MODULE
1206
SERVO CONTROLLER
Cap
Cap
Sample
Sample
SrvCNTRLAdr[9:0]
APBSnapReg
1202B
APB Bus Read Data Mux
1210 DCDR
DSPSnapReg
1201B
DSP Read Data Mux
SRVCNTRLWData[31:00]
PADDR[11:02]
Memory Mapped Reg Write Enablee
Read Data Mux
Read Data Mux
SERVO CONTROLLER MEMORY MAPPED REGISTERS
1209
1202A
1202
1201A
1201
SERVO STATE MACHINES
READ CHANNEL BIT SERIAL LINE
HDIC BIT SERIAL LINE
SV BIT SERIAL LINE
VCO COUNTER
MULTIRATE CONTROL
1200
HDICSERIALLINE
RDCHNLSERIAL LINE
SVSERIALLINE
216

Figure 12B

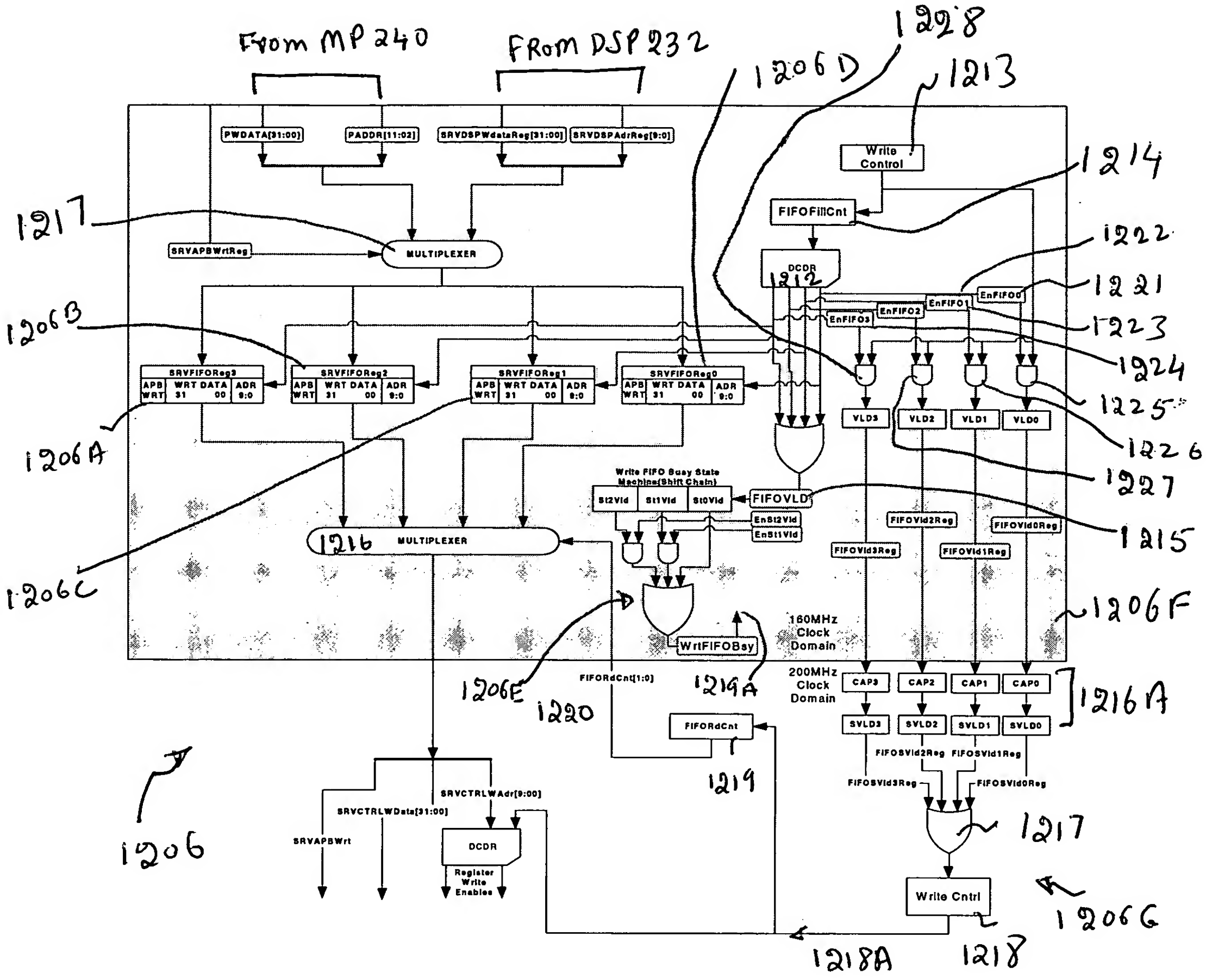

From MP 240
From DSP 232
1208
1206D
1213
PWDATA[31:00]
PADDR[11:02]
SRVDSPWdataReg[31:00]
SRVDSPAdrReg[9:0]
Write Control
1214
1217
FIFOFillCnt
SRVAPBWrtReg
MULTIPLEXER
1222
DCDR
1212
1221
EnFIFO0
EnFIFO1
EnFIFO2
EnFIFO3
1206B
1223
1224
SRVFIFOReg3
SRVFIFOReg2
SRVFIFOReg1
SRVFIFOReg0
APB WRT
WRT DATA 31 00
ADR 9:0
1225
VLD3
VLD2
VLD1
VLD0
1206A
1226
1227
Write FIFO Busy State Machine(Shift Chain)
St2Vld
St1Vld
St0Vld
FIFOVLD
EnSt2Vld
EnSt1Vld
FIFOVld2Reg
FIFOVld0Reg
1215
1216 MULTIPLEXER
FIFOVld3Reg
FIFOVld1Reg
1206C
1206F
160MHz Clock Domain
WrtFIFOBsy
1206E
FIFORdCnt[1:0]
1219A
200MHz Clock Domain
CAP3
CAP2
CAP1
CAP0
1216A
1220
FIFORdCnt
SVLD3
SVLD2
SVLD1
SVLD0
1219
FIFOSVld2Reg
FIFOSVld1Reg
FIFOSVld3Reg
FIFOSVld0Reg
SRVCTRLWAdr[9:00]
SRVCTRLWData[31:00]
1217
1206
SRVAPBWrt
DCDR
Register Write Enables
Write Cntrl
1206G
1218A
1218

Figure 12C

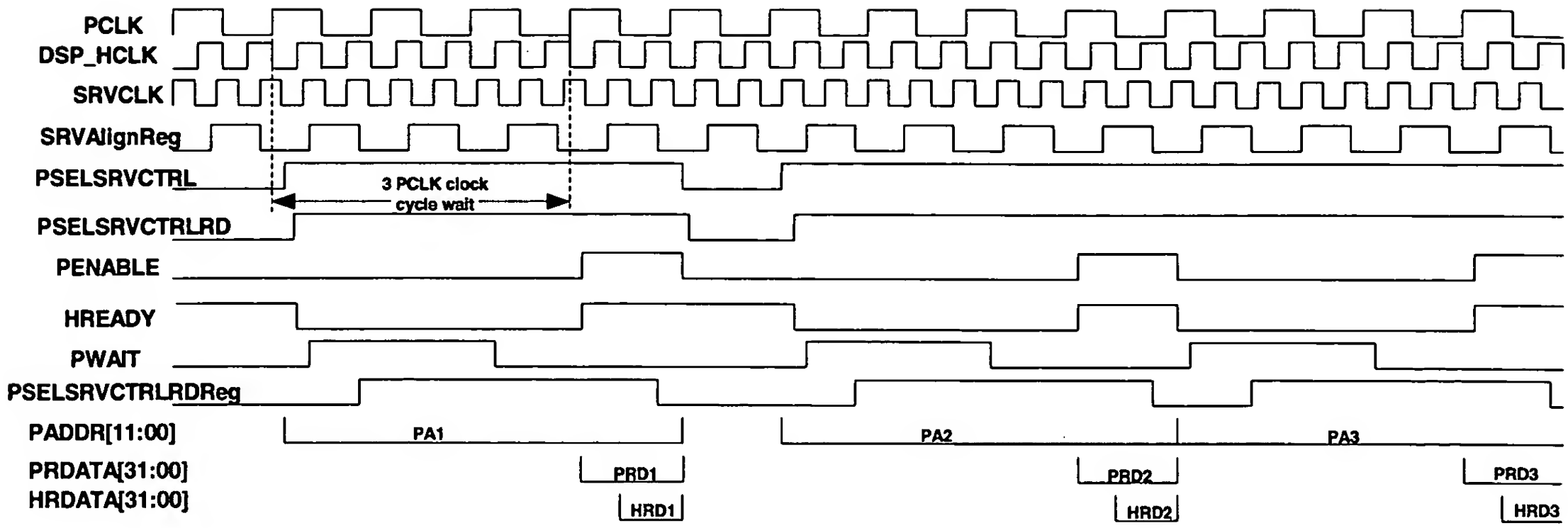
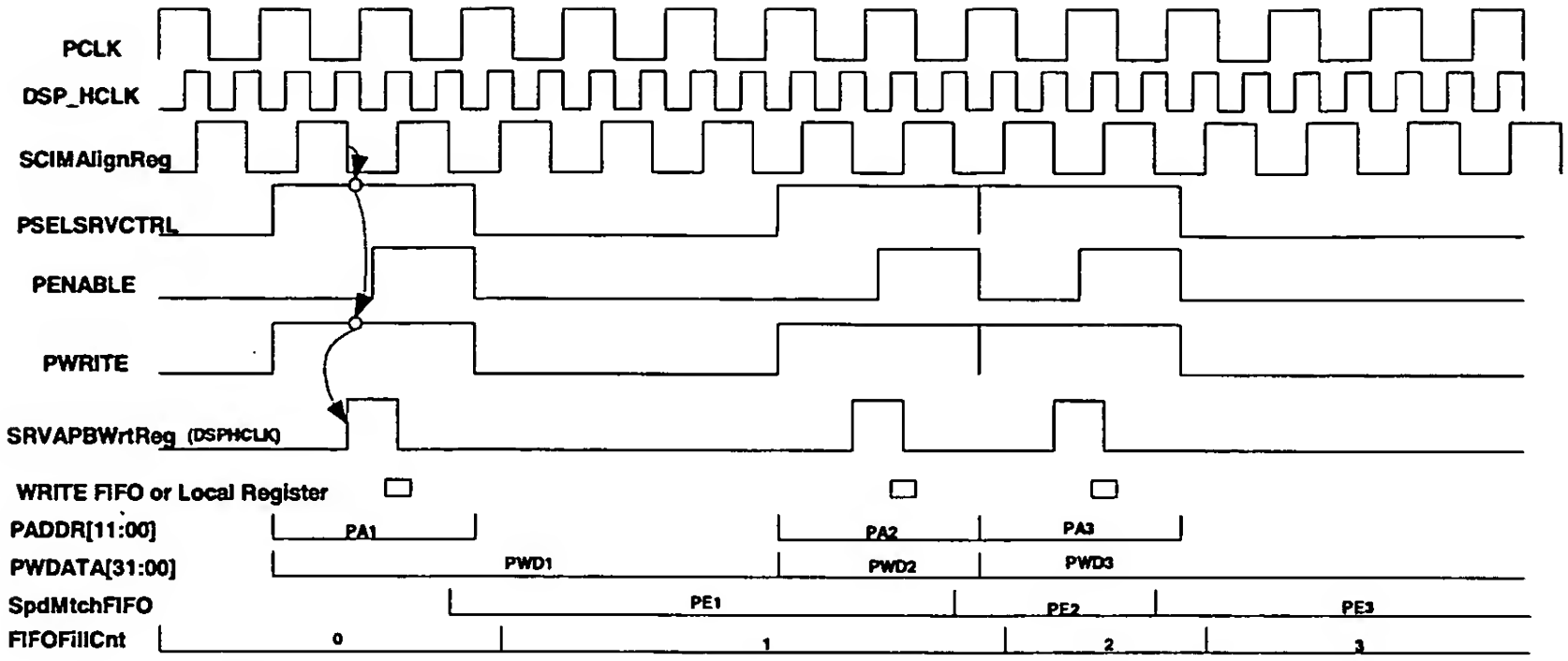

PCLK
DSP_HCLK
SRVCLK
SRVAlignReg
PSELSRVCTRL
3 PCLK clock cycle wait
PSELSRVCTRLRD
PENABLE
HREADY
PWAIT
PSELSRVCTRLRDReg
PADDR[11:00]
PA1
PA2
PA3
PRDATA[31:00]
PRD1
PRD2
PRD3
HRDATA[31:00]
HRD1
HRD2
HRD3

Figure 13



NOTE:

1. SRVAPBWrtData, data input to SRVAPBWrtReg = PSELSRVCTRL & PWRITE & ~PENABLE & SCIMAlignReg

Figure 14

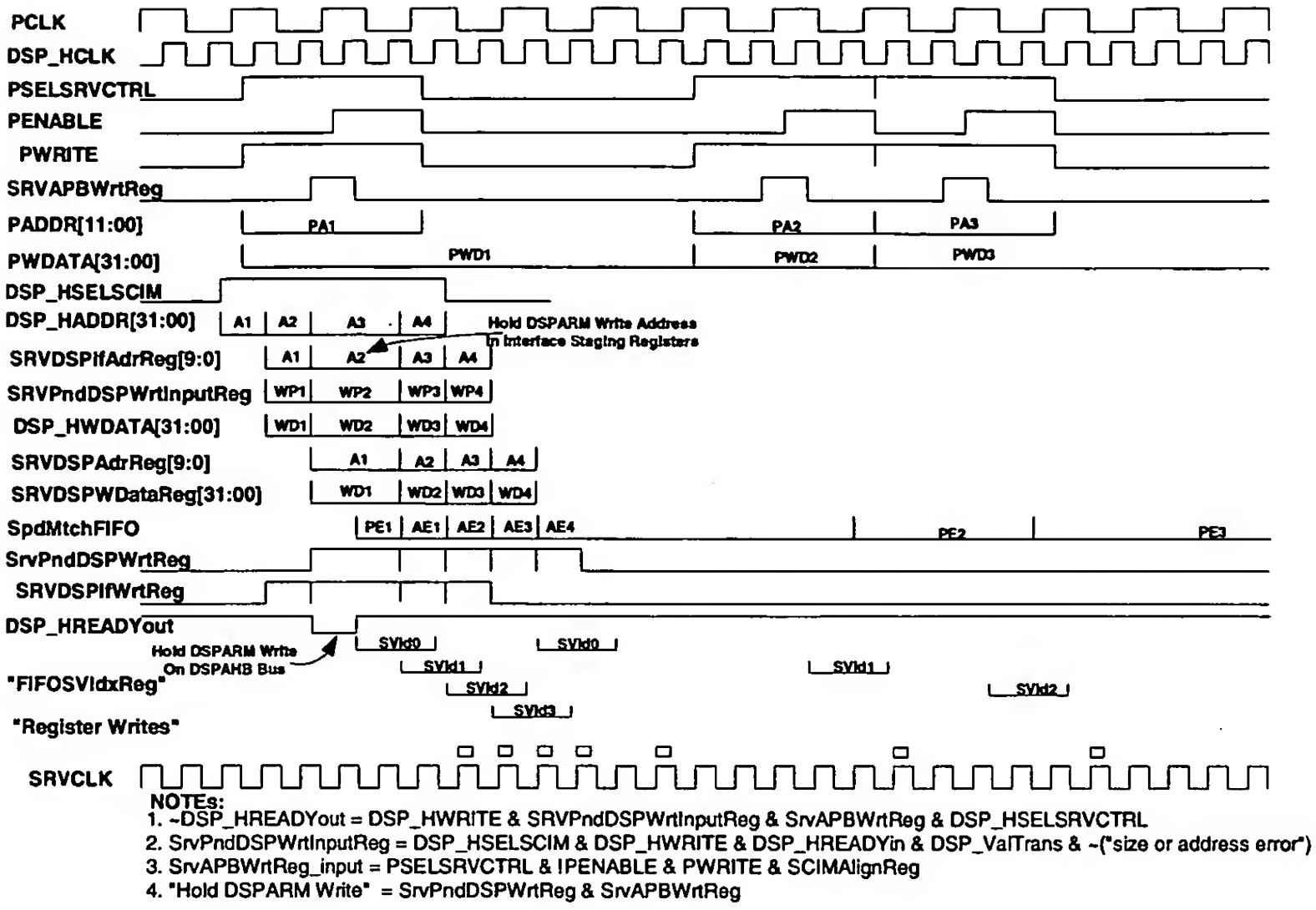

PCLK
DSP_HCLK
PSELSRVCTRL
PENABLE
PWRITE
SRVAPBWrtReg
PADDR[11:00]
PA1
PA2
PA3
PWDATA[31:00]
PWD1
PWD2
PWD3
DSP_HSELSCIM
DSP_HADDR[31:00]
A1
A2
A3
A4
Hold DSPARM Write Address
In Interface Staging Registers
SRVDSPIfAdrReg[9:0]
SRVPndDSPWrtInputReg
WP1
WP2
WP3
WP4
DSP_HWDATA[31:00]
WD1
WD2
WD3
WD4
SRVDSPAdrReg[9:0]
SRVDSPWDataReg[31:00]
SpdMtchFIFO
PE1
AE1
AE2
AE3
AE4
PE2
PE3
SrvPndDSPWrtReg
SRVDSPIfWrtReg
DSP_HREADYout
Hold DSPARM Write
On DSPAHB Bus
"FIFOSVIdxReg"
SVId0
SVId1
SVId2
SVId3
"Register Writes"
SRVCLK
NOTEs:
1. ~DSP_HREADYout = DSP_HWRITE & SRVPndDSPWrtInputReg & SrvAPBWrtReg & DSP_HSELSRVCTRL
2. SrvPndDSPWrtInputReg = DSP_HSELSCIM & DSP_HWRITE & DSP_HREADYin & DSP_ValTrans & ~("size or address error")
3. SrvAPBWrtReg_input = PSELSRVCTRL & !PENABLE & PWRITE & SCIMAlignReg
4. "Hold DSPARM Write" = SrvPndDSPWrtReg & SrvAPBWrtReg

Figure 15

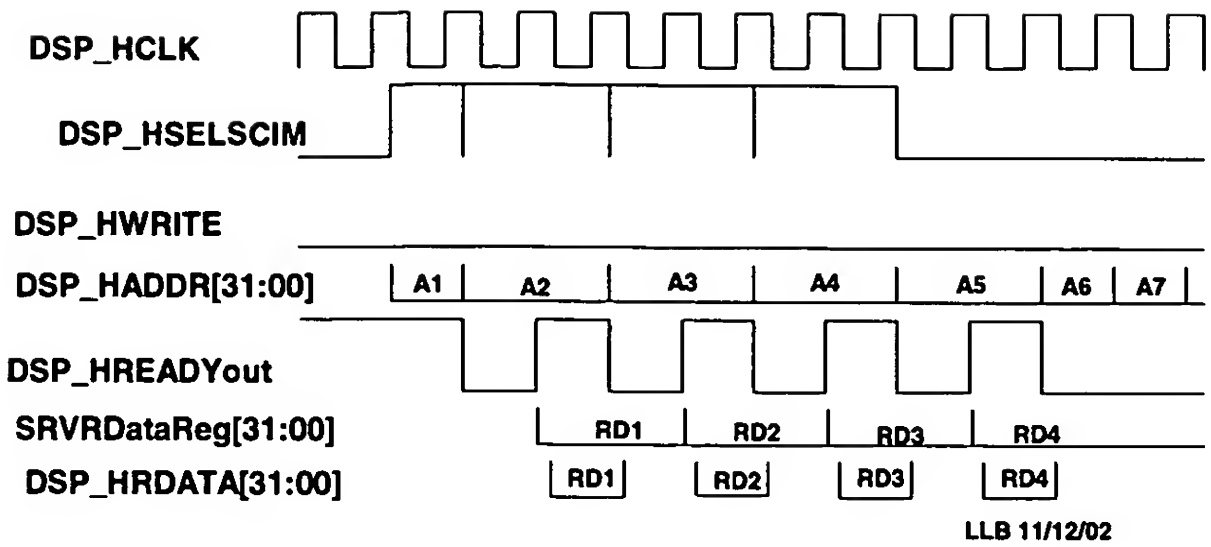


Figure 16

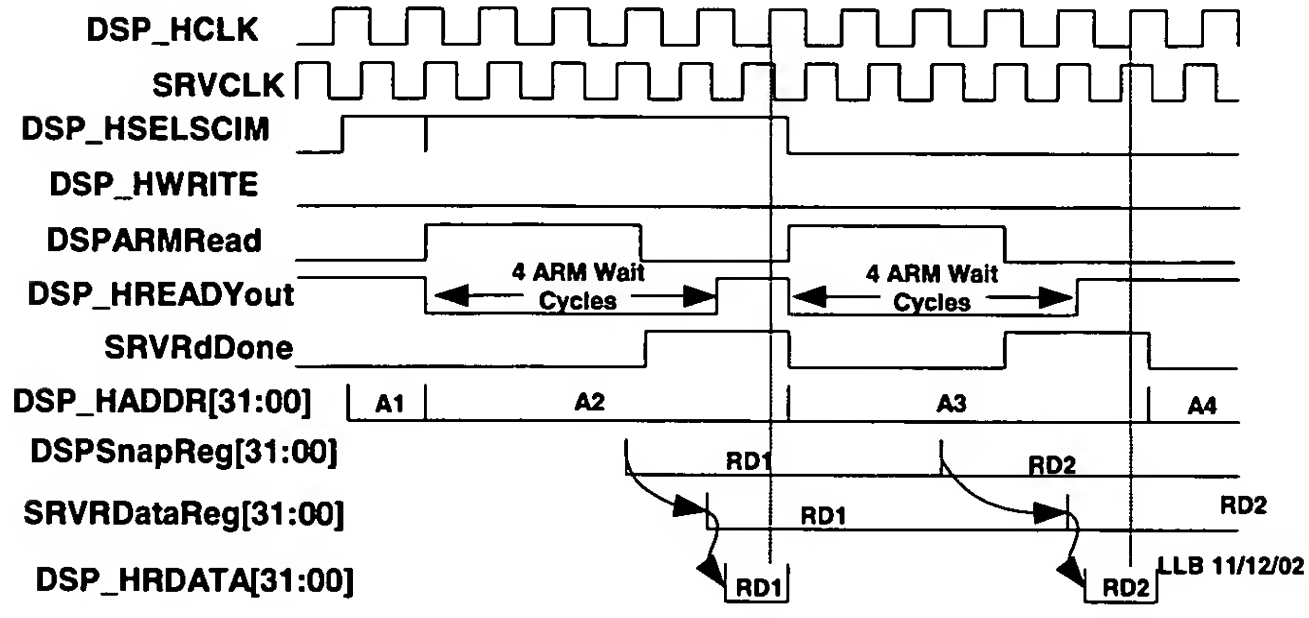


Figure 17

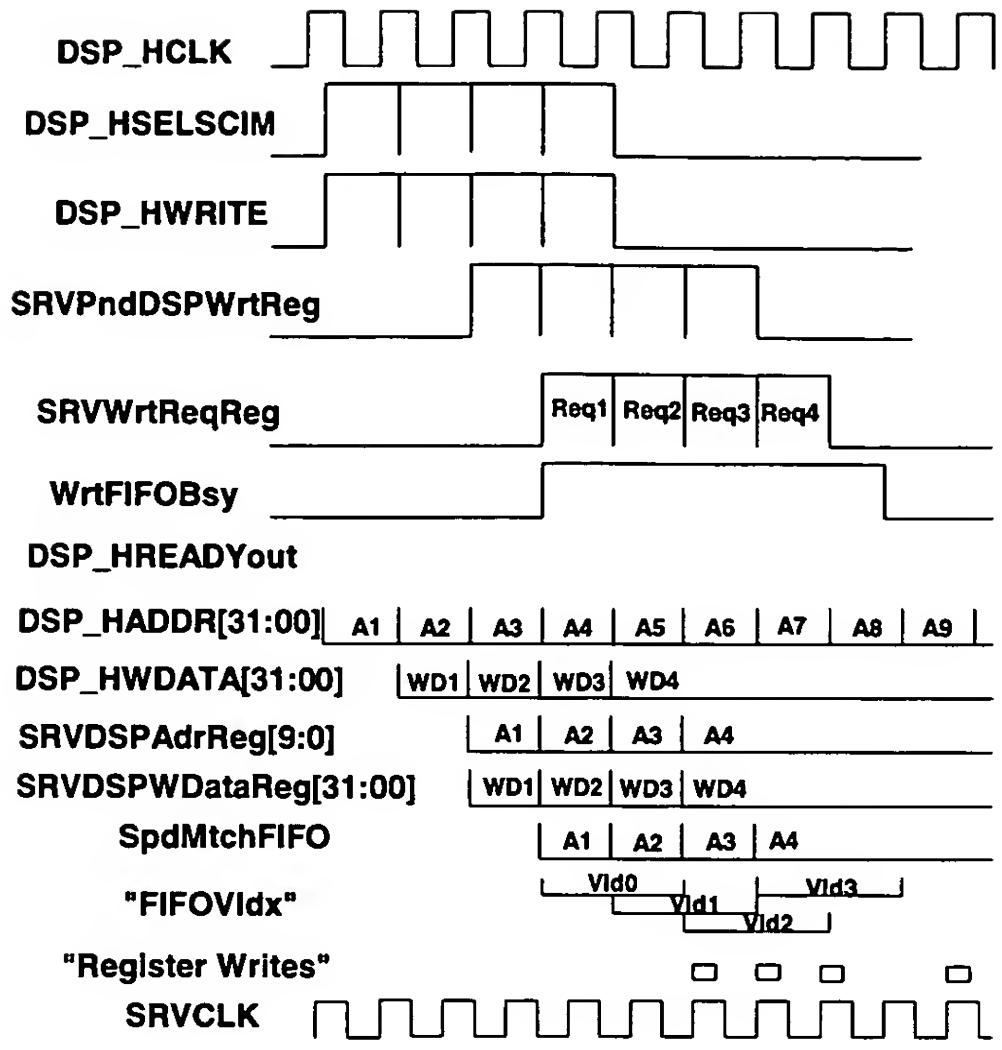

DSP_HCLK
DSP_HSELSCIM
DSP_HWRITE
SRVPndDSPWrtReg
SRVWrtReqReg
Req1
Req2
Req3
Req4
WrtFIFOBsy
DSP_HREADYout
DSP_HADDR[31:00]
A1
A2
A3
A4
A5
A6
A7
A8
A9
DSP_HWDATA[31:00]
WD1
WD2
WD3
WD4
SRVDSPAdrReg[9:0]
A1
A2
A3
A4
SRVDSPWDataReg[31:00]
WD1
WD2
WD3
WD4
SpdMtchFIFO
A1
A2
A3
A4
"FIFOVldx"
Vld0
Vld1
Vld2
Vld3
"Register Writes"
SRVCLK

Figure 18

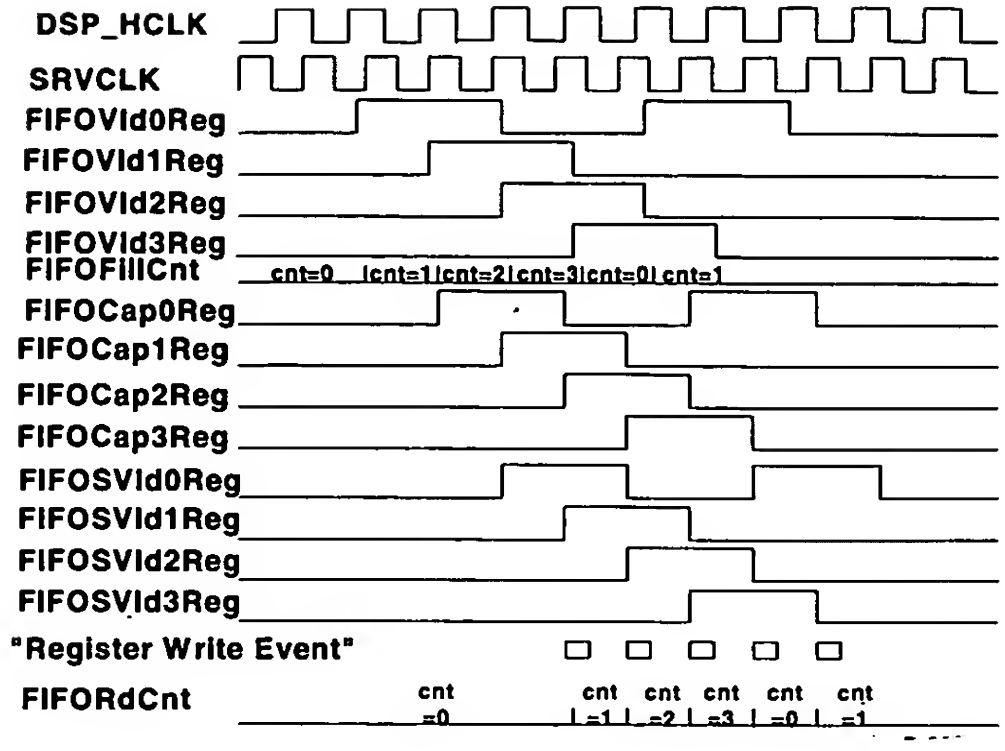

DSP_HCLK
SRVCLK
FIFOVld0Reg
FIFOVld1Reg
FIFOVld2Reg
FIFOVld3Reg
FIFOFillCnt
cnt=0
cnt=1
cnt=2
cnt=3
cnt=0
cnt=1
FIFOCap0Reg
FIFOCap1Reg
FIFOCap2Reg
FIFOCap3Reg
FIFOSVld0Reg
FIFOSVld1Reg
FIFOSVld2Reg
FIFOSVld3Reg
"Register Write Event"
FIFORdCnt
cnt =0
cnt =1
cnt =2
cnt =3
cnt =0
cnt =1

Figure 19

Conflict		APB 208 Waits	DSPAHB Bus 233 Waits
APB 208	DSP 232		
Write	Write	No	Yes- 1 wait cycle
Read local register	NA	No	No
NA	Read local register	No	Yes - 1 DSP_HCLK clock cycle
Read Servo Controller register	NA	Yes - 1 PCLK clock cycle	No
NA	Read Servo Controller register	No	Yes - 3 DSP_HCLK clock cycles

Figure 20

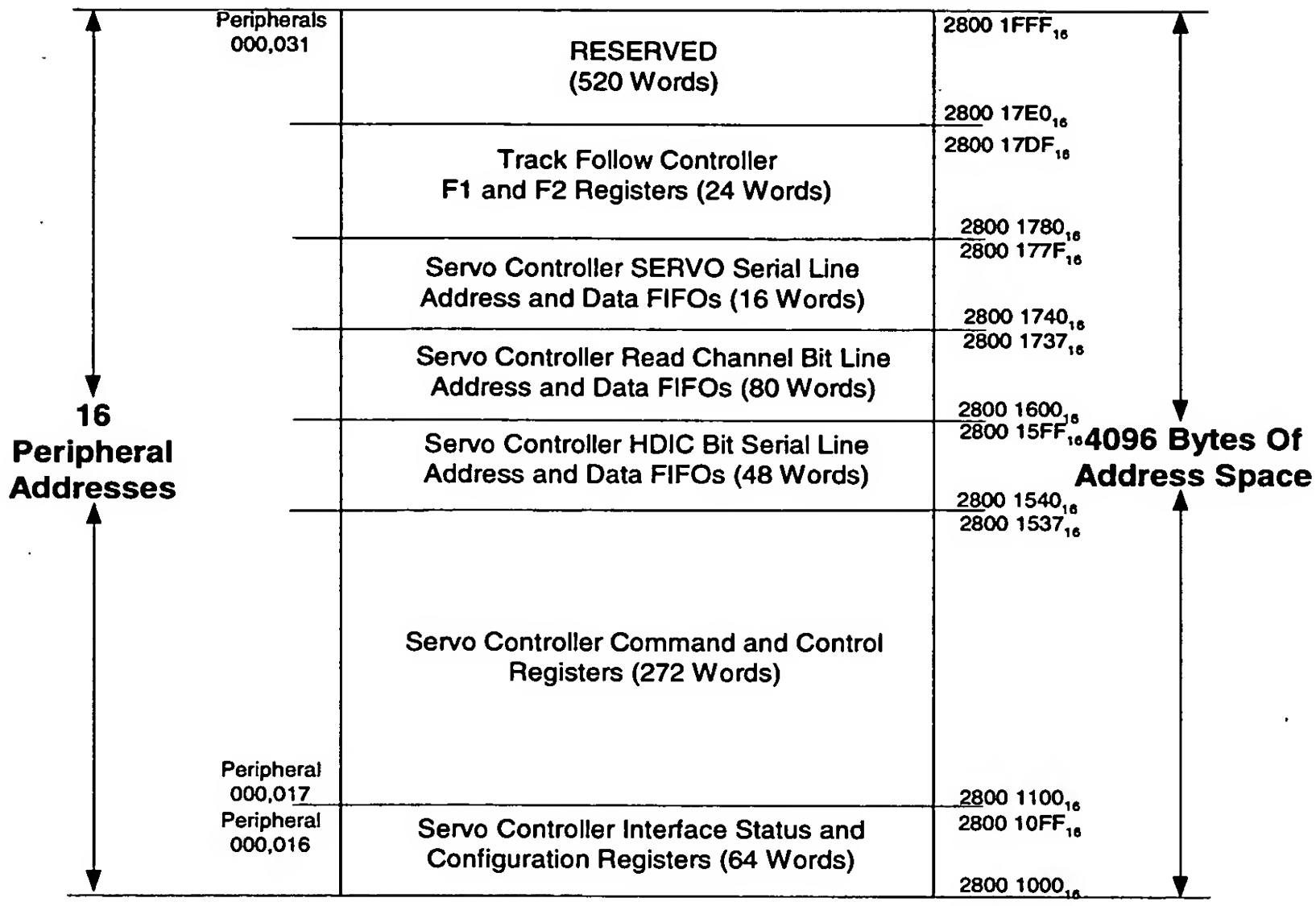

Peripherals
000,031
RESERVED
(520 Words)
2800 1FFF16
2800 17E016
Track Follow Controller
F1 and F2 Registers (24 Words)
2800 17DF16
2800 178016
Servo Controller SERVO Serial Line
Address and Data FIFOs (16 Words)
2800 177F16
2800 174016
Servo Controller Read Channel Bit Line
Address and Data FIFOs (80 Words)
2800 173716
2800 160016
Servo Controller HDIC Bit Serial Line
Address and Data FIFOs (48 Words)
2800 15FF16
2800 154016
Servo Controller Command and Control
Registers (272 Words)
2800 153716
Peripheral
000,017
2800 110016
Peripheral
000,016
Servo Controller Interface Status and
Configuration Registers (64 Words)
2800 10FF16
2800 100016
16
Peripheral
Addresses
4096 Bytes Of
Address Space

Figure 21

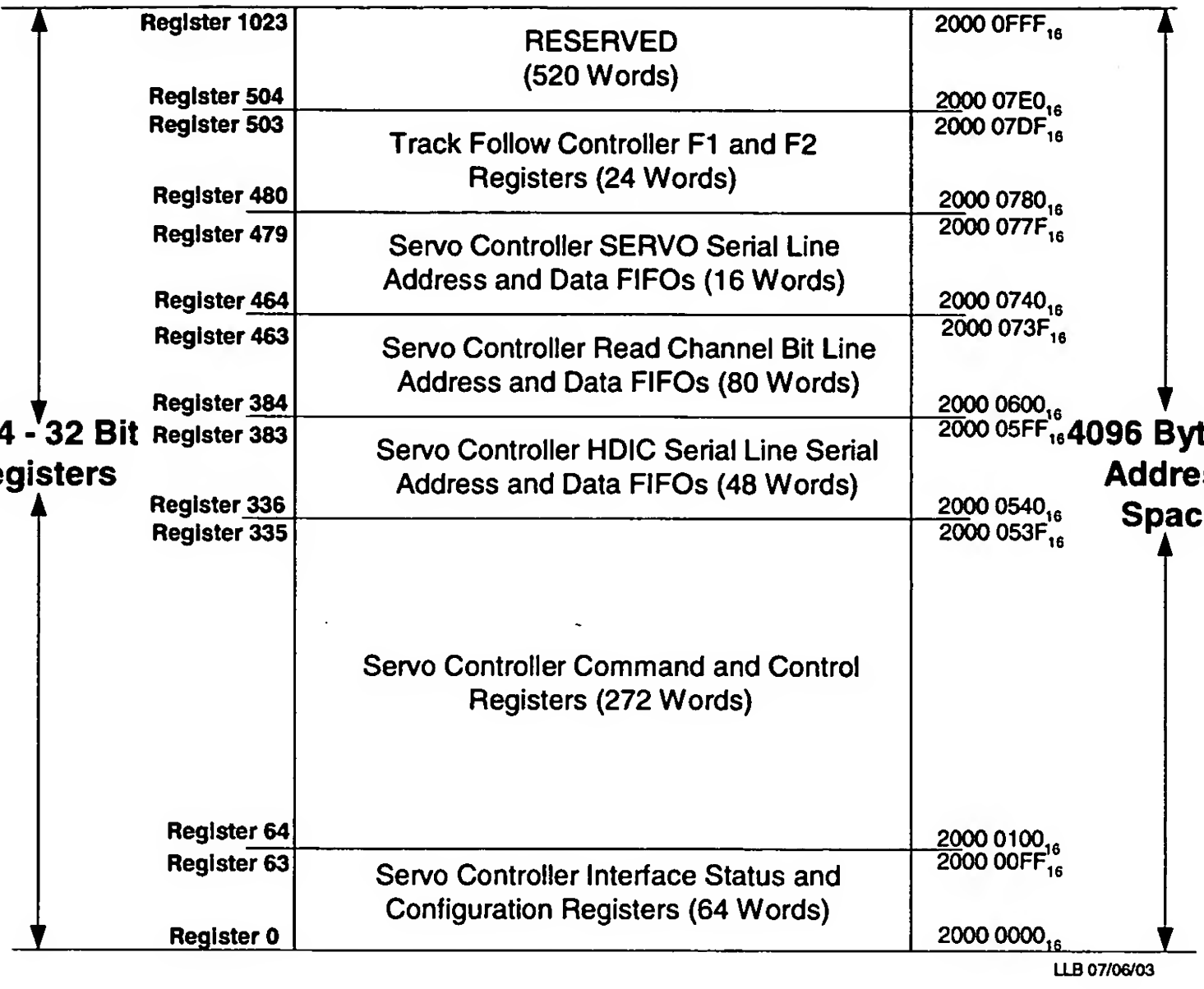

Register 1023
RESERVED
(520 Words)
Register 504
2000 0FFF16
2000 07E016
Register 503
Track Follow Controller F1 and F2
Registers (24 Words)
Register 480
2000 07DF16
2000 078016
Register 479
Servo Controller SERVO Serial Line
Address and Data FIFOs (16 Words)
Register 464
2000 077F16
2000 074016
Register 463
Servo Controller Read Channel Bit Line
Address and Data FIFOs (80 Words)
Register 384
2000 073F16
2000 060016
1024 - 32 Bit
Registers
Register 383
Servo Controller HDIC Serial Line Serial
Address and Data FIFOs (48 Words)
Register 336
2000 05FF16
2000 054016
4096 Bytes of
Address
Space
Register 335
2000 053F16
Servo Controller Command and Control
Registers (272 Words)
Register 64
2000 010016
Register 63
Servo Controller Interface Status and
Configuration Registers (64 Words)
Register 0
2000 00FF16
2000 000016
LLB 07/06/03

Figure 22

Bits	Function
[04]	**SCIARMADCInt** – SCI 211 sets (one clock cycle pulse) this bit when conversion data is available from ADCDAC 213. Setting this bit generates a level sensitive interrupt (SRVCTRLARMINT) to Processor 240.
[03]	**ARMWrtFIFOBsy** - Write FIFOBsy signal bit indicates that a pending write is still in the Speed Matching Write FIFO 1206.
[02]	**SCIARMSmphCflt** – When SCI 211 sets this bit, it asserts the PADREXCPT signal to APB Bridge 235 indicating that Processor 240 attempted a register write (any register other than the SCISemaphoreReg or this register) when DSP 232 had ownership of the SCIHdwSmphr bit (hardware semaphore, SCISemaphoreReg[00]).
[01]	**SCIARMAdrExcpt** – SCI 211 sets (one clock cycle pulse) this bit when an Address Exception is detected on Processor 240 access to SC 216 register.
[00]	**SCIARMInt** – SCI 211 generates a level sensitive interrupt (SRVCTRLARMINT) to Processor 240 when SC 216 asserts the hardware interrupt signal, SRVCARMINT.

Figure 23

Bits	Function
[31:04]	Reserved
[03]	**DSPWrtFIFOBsy** - Write FIFOBsy bit indicates that a pending write is still in the Speed Matching Write FIFO 1206.
[02]	**SCIDSPSmphCflt** – SCI 211 sets (one clock cycle pulse) this bit when DSP 232 attempts a register write (any register except SCISemaphoreReg and this register) without having ownership of SCISemaphoreReg[00], the hardware semaphore bit SCIHdwSmphr. Setting this bit generates a level sensitive interrupt (SRVCTRLDSPINT) to the DSP Interface Module 210.
[01]	**SCIDSPAdrExcpt** – SCI 216 sets (one clock cycle pulse) this bit when an Address Exception is detected on DSP 232 access to SC 216. Setting this bit generates a level sensitive interrupt (SRVCTRLDSPINT) to DSP 232.
[00]	**SCIDSPInt** – SCI 211 generates a level sensitive interrupt (SRVCTRLDSPINT) to DSP 232 when SC 216 asserts the hardware interrupt signal, SRVCDSPINT.

Figure 24

)

Bits	Function
[31:04]	Reserved
[03:01]	**SCISftSmphr[02:00]** – SCI 211 Soft Semaphores that provide procedural interlocks only. These semaphores do not provide a hardware interlock (see description below).
[00]	**SCIHdwSmphr** - SCI 211 Hardware Semaphore that provides a hardware interlock. When APB 208 acquires the semaphore the DSP 232 cannot execute a write access to any register in SC 216 or SCI 211 except the Semaphore Register or its respective status register. The reverse is true when DSP 232 owns this semaphore.

Figure 24A

Bit Write Data	Alternate Bus State	Current Bus State	Results	Comment
0	0	0	0	Write bit position = 0, no action.
0	1	0	0	Write bit position = 0, no action.
1	1	0	0	Attempted to acquire semaphore while the alternate bus has ownership.
1	0	0	1	Attempt to acquire semaphore is successful.
0	0	1	1	Write bit position = 0, no action. Semaphore remains undisturbed.
1	0	1	0	Clear semaphore operation.

Figure 25

Bit Write Data	Alternate Bus State	Current Bus State	Results	Comment
0	0	0	0	Write bit position = 0, no action.
0	1	0	0	Write bit position = 0, no action.
1	1	0	0	Attempted to acquire semaphore while the alternate bus has ownership.
1	0	0	1	Attempt to acquire semaphore is successful.
0	0	1	1	Write bit position = 0, no action. Semaphore remains undisturbed.
1	0	1	0	Clear semaphore operation.

Figure 25

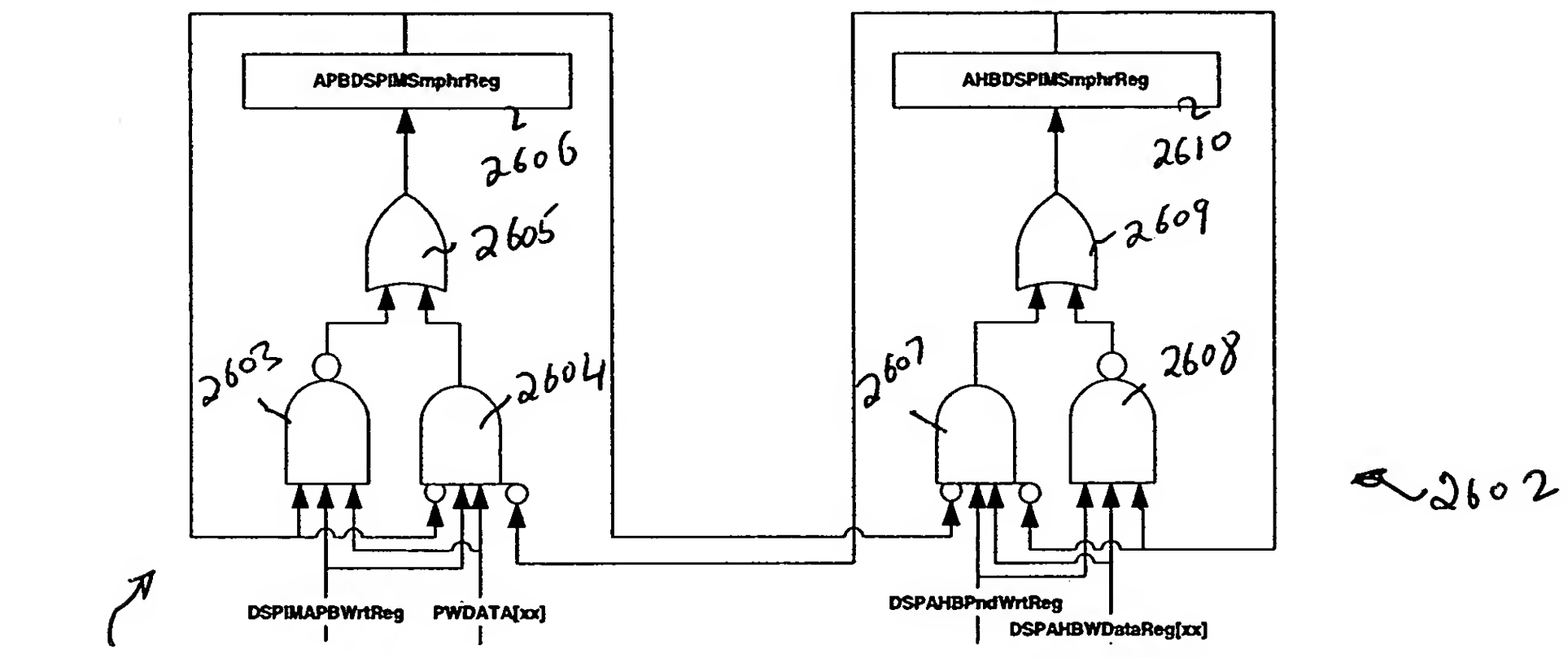


Figure 26

Bits	Function
[31:10]	Reserved
[09]	**EnAddDSPWt** - Enable Additional DSPARM Wait bit may be set (=1) to provide an additional DSP 232 wait cycle when reading a memory mapped register from SC 216. When this bit = 0 waits states on reads to SC 216 memory mapped registers = 3. When this bit = 1 waits states on reads to SC 216 memory mapped registers = 4.
[08]	**EnDSPFrstADCCnvCmpltInt** – Enable DSP 232 First ADC Conversion Complete Interrupt bit enables SCI 211 to send an interrupt (ADCCNVTDSPINT) to DSP 232 each time a conversion completes as the result of a Servo Frame (ADCFisrtCnvt signal from SC 216)
[07]	**EnDSPADCCnvCmpltInt** – Enable DSP 232 ADC Conversion Complete Interrupt bit enables SCI 211 to send an interrupt (ADCCNVTDSPINT) to DSP 232 each time a conversion completes.
[06]	**EnARMFrstADCCnvCmpltInt** – Enable Processor 240 First ADC Conversion Complete Interrupt bit enables SCI 211 to send an interrupt (SRVCTRLARMINT) to Interrupt Controller 207 each time a conversion completes as the result of a Servo Frame (ADCFisrtCnvt signal from SC 216)
[05]	**EnARMADCCnvCmpltInt** – Enable Processor 240 ADC Conversion Complete Interrupt bit enables the SCI Controller Interface 211 to send a Processor 240 interrupt (SRVCTRLARMINT) signal Interrupt Controller 207 each time a conversion completes.
[04]	**EnOnTrackSMVal** – When this bit is set the Enable On Track Servo Machine values (DACval[9:0] are accepted each time a DAC Advance signal (DACAdvn) is sent rather than from the DAC FIFO values.
[03]	**EnDSPPSDataInt** –. Enable DSP Position Data Interrupt enables SCI 211 to send interrupt SRVDSPARMPSDATAINT signal to DSP 232 each time SC 216 sends SRVPsDataInt.
[02]	**EnSCIDSPInt** – Enables SCI 211 to generate an interrupt (SRVCTRLDSPARMINT) signal to DSP 232. When this bit = 0 it blocks all sources of interrupt generation. When this bit = 1 enable all sources to generate an interrupt.
[01]	**EnADCCnvsn** – When this bit = 0 the ADC converter 213 is powered down and may not generate conversion data or interrupts.
[00]	**EnDSPIntfc** – Enable the DSPARM Interface (DSPAHB Bus) bit may be set or cleared by APB 208. DSP 232 may clear this bit, but it cannot set this bit as it gains access to SCI 211 with MP 240 setting this bit. When this bit is set, DSP 232 has read/write access. When this bit is clear DSP 232 does not have write access to any registers except the SCIDSPStatusReg. An access attempted by DSP 232 without this bit set will result in an Address Exception interrupt being returned to DSP 232.

Figure 27

Servo Controller Interface MultiRate ADC Mask Register (SCIMultRtADCMskReg)

Bits	Function
[31:07]	Reserved
[06:02]	**ADC Mask** bits specify the ADC 213 channel values that will be retained (1 mask bit for each channel) after a requested conversion with the exception of the conversion taken at SBD time (ADCcnt). All 5-channel conversion values are retained at SBD time (ADCFirstCnvt).
[01:00]	**Multiple Rate Control** – This two bit field specifies the number of valid DAC Multiple Rate FIFO entries. 00 = Disabled, 01 = 1 X Multiple Rate, 10 = 2 X Multiple Rate, 11 = 4 X Multiple Rate.

Figure 28

Servo Controller Interface DAC MultiRate 0 Register (SCIDACMultRt0Reg)

Bits	Function
[09:00]	**Entry 0 of the DAC Multi RateFIFO** - This value is specified by either Processor 240 or DSP 232 for the first value provided to the DAC 213.

Figure 29

Servo Controller Interface DAC MultiRate 1 Register (SCIDACMultRt1Reg)

Bits	Function
[09:00]	**Entry 1 of the DAC MultiRate FIFO** - This value is specified by either Processor 240 or DSP 232 for the second value provided to the DAC 213 during a Multi Rate operation.

Figure 30

Servo Controller Interface DAC MultiRate 2 Register (SCIDACMultRt2Reg)

Bits	Function
[09:00]	**Entry 2 of the DAC MultiRate FIFO** - This value is specified by either processor 240 or DSP 232 for the third value provided to the DAC during a Multi Rate operation.

Figure 31

Servo Controller Interface DAC MultiRate 3 Register (SCIDACMultRt3Reg)

Bits	Function
[09:00]	**Entry 3 of the DAC MultiRate FIFO** - This value is specified by either Processor 240 or the DSP 232 for the third value provided to the DAC during a Multi Rate operation.

Figure 32

Servo Controller Interface ADC Channel 0 Register (SCIADCChn0Reg)

Bits	Function
[09:00]	**Converted value from ADC channel 0** - This value is from a channel 0 conversion of an attached analog signal.

Figure 33

Servo Controller Interface ADC Channel 1 Register (SCIADCChn1Reg)

Bits	Function
[09:00]	**Converted value from ADC channel 1** - This value is from a channel 1 conversion of an attached analog signal.

Figure 34

Servo Controller Interface ADC Channel 2 Register (SCIADCChn2Reg)

Bits	Function
[09:00]	**Converted value from ADC channel 2** - This value is from a channel 2 conversion of an attached analog signal.

Figure 35

Servo Controller Interface ADC Channel 3 Register (SCIADCChn3Reg)

Bits	Function
[09:00]	**Converted value from ADC channel 3**- This value is from a channel 3 conversion of an attached analog signal..

Figure 36

Servo Controller Interface ADC Channel 4 Register (SCIADCChn4Reg)

Bits	Function
[09:00]	**Converted value from ADC channel 4** - This value is from a channel 4 conversion of an attached analog signal.

Figure 37

Servo Controller Interface Positioning Error Signal Data Array Registers

Bits	Function
[23:00]	**Positioning Error Data** – These 24 bits contain the Positioning Error Data value of each register array entry.

Figure 38